Enhanced OS-9 for the STPC

Version 1.2

MICROWARE™

Intelligent Products For A Smarter World

Copyright and Publication Information

Copyright ©2000 Microware Systems Corporation. All Rights Reserved. Reproduction of this document, in part or whole, by any means, electrical, mechanical, magnetic, optical, chemical, manual, or otherwise is prohibited, without written permission from Microware Systems Corporation.

This manual reflects version 1.2 (of Enhanced OS-9 for X86.

Revision: A
Publication date: February 2000

Disclaimer

The information contained herein is believed to be accurate as of the date of publication. However, Microware will not be liable for any damages including indirect or consequential, from use of the OS-9 operating system, Microware-provided software, or reliance on the accuracy of this documentation. The information contained herein is subject to change without notice.

Reproduction Notice

The software described in this document is intended to be used on a single computer system. Microware expressly prohibits any reproduction of the software on tape, disk, or any other medium except for backup purposes. Distribution of this software, in part or whole, to any other party or on any other system may constitute copyright infringements and misappropriation of trade secrets and confidential processes which are the property of Microware and/or other parties. Unauthorized distribution of software may cause damages far in excess of the value of the copies involved.

For additional copies of this software/documentation, or if you have questions concerning the above notice, please contact your OS-9 supplier.

Trademarks

OS-9, OS-9000, DAVID, and MAUI are registered trademarks of Microware Systems Corporation. SoftStax, FasTrak, UpLink, MICROWARE TECH-CHECK, and Hawk are trademarks of Microware Systems Corporation. All other product names referenced herein are either trademarks or registered trademarks of their respective owners.

Address

Microware Systems Corporation
1500 N.W. 118th Street
Des Moines, Iowa 50325
515-223-8000

Table of Contents

Chapter 1: Installing and Configuring Enhanced OS-9 **5**

6 Introduction
7 Hardware Description
7 Applicable Documents
8 Product Support
8 Installing OS-9 on the Windows Host Computer
8 Configuring the Hardware
8 Jumper Settings
9 SIMM Modules
9 Disk Drives
9 Monitor
9 Keyboard/Mouse
9 Ethernet Support
10 CMOS Settings
11 Installing OS-9 on the STPC System Board
11 Jumpers and CMOS Configuration
12 Making a Boot Floppy
17 Preparing the Hard Disk
19 Advanced Configurations
28 Finishing the Hard Disk Configuration
29 MAUI Graphics Support
32 Cross-hosted Software Development Using Microware Hawk
32 Starting Hawk on the Target
32 Running MAUI Graphics
33 Running the Hawk Tutorial
33 Project File Creation
34 Customizing the Component
38 Building the Project

39 Loading and Executing the HomeWatch Application
39 Security
40 Modifying the Key Code

Product Discrepancy Report **41**

Chapter 1: Installing and Configuring Enhanced OS-9

This chapter describes installing and configuring Enhanced OS-9 for X86. It includes the following sections:

- **Introduction**
- **Hardware Description**
- **Product Support**
- **Cross-hosted Software Development Using Microware Hawk**

Introduction

This manual provides information to help you get started with OS-9 for the STPC. OS-9 is an architecturally advanced, high performance real-time operating system available for the IBM/Motorola, Intel, SuperH, MIPS and ARM/StrongARM microprocessor families. Refer to the documentation on the Enhanced OS-9 for the STPC CD for additional OS-9 product information.

Enhanced OS-9 for the STPC includes the OS-9 stand-alone microkernel for X86; Hawk, Microware's cross-hosted development solution; a copy of Microware's new configuration builder tool, graphics, networking and complete on-line documentation.

The host installation requires:

- Microsoft Windows 95, 98 or NT
- Pentium Processor
- 32 MB RAM
- 250 - 350 MB Free Disk Space
- CD-ROM Drive
- Network Card

The STPC installation requires:

- Evaluation board with STPC Client, Consumer or Industrial
- VGA Monitor
- IDE Hard Disk
- Floppy Drive
- OS-9 Supported Network Card
- Keyboard, Mouse is Optional

Hardware Description

The GloriaBD and ConsumerBD are high-performance personal computer system boards based on the STPC Client and Consumer microprocessors.

Applicable Documents

Refer to *Getting Started with Enhanced OS-9 for X86* for additional information in configuring and installing OS-9 on your X86/Pentium based target. Refer to the full OS-9 documentation set on the product CD for additional information on the OS-9 operating system. Additional information on the STPC Evaluation Kit is included in the user manual shipped with your evaluation hardware.

Product Support

Microware stands behind its products with an experienced staff of Technical Support engineers, available to assist you with your development projects from 8:00 AM to 6:00 PM CST. Whether you have installation, configuration, basic usage, or advanced questions, our team is poised to take on the challenge.

Telephone support is available by calling 515-224-0458. You may also send your questions to support@microware.com or post your questions on our bulletin board at bbs.microware.com.

Product release notes are available at http://www.microware.com/x86relnotes.

Installing OS-9 on the Windows Host Computer

Microware's Enhanced OS-9 for the STPC must be installed on your Windows 95/98 or NT computer, prior to installing OS-9 on your target computer. Insert the OS-9 CD-ROM into your computer and select one of the installation options from the autorun menu. Follow the on-line instructions to complete the install.

The installation creates an "OS-9 for X86" program folder and includes full documentation, Microware's Hawk development environment, Microware's Configuration Wizard and the OS-9 for X86 binary images.

The package may be removed by selecting "Uninstall OS-9 for X86" from the program menu.

Configuring the Hardware

Jumper Settings

Refer to the Hardware Installation section of the board manual when setting the jumpers.

SIMM Modules

Refer to the Hardware Installation section of the board manual when installing the SIMM modules. OS-9 will run with as little as 2 MB of RAM, however, it may be more convenient to install additional memory when developing and testing graphic intensive applications.

Disk Drives

While OS-9 can be configured as a ROM based system, these instructions assume that a floppy drive and IDE hard disk will be used for initial development and testing. Connect the floppy and IDE drives to the motherboard, using the FDC1 and IDE1 board connectors.

Monitor

Connect a VGA monitor to the VA1 connector on the board. You may optionally connect an NTSC monitor to the P2 connector on the board.

Keyboard/Mouse

Connect a PS/2 style keyboard and mouse to the KB1 connectors on the board.

Ethernet Support

Development work for will be facilitated with the addition of an ISA or PCI network card. The network card must be supported by OS-9. Additional cards are constantly being added, so check bbs.microware.com for the latest list. Current Ethernet support includes the following devices:

- 3COM PCI EtherLink XL
- 3COM PCI EtherLink III
- 3COM ISA EtherLink III
- SMC Ultra 83C790

- SMC 8390
- 3COM EtherLink III PC CARD
- DEC 21140
- AM79C961A & AM79C973
- NE2000

Commonly available cards, supported by OS-9, include:

- Ready LINK COMPEX RL2000-PCI (use ne2000 Configuration Wizard setting)
- Etherlink XL PCI 3C900-TPO (use 3COM PCI EtherLink XL Configuration Wizard setting)
- Asanté Fast 10/100 PCI Adapter for the Mac & PC (use Dec 21140 Configuration Wizard setting)

For some Ethernet cards, the I/O base address and interrupt settings must be configured on the card to match the settings used by OS-9. A setup disk, provided with the network card, may be needed to configure the card to the correct settings. The default settings for an NE2000 card are I/O Base 0x340 and IRQ 9. Refer to the Ethernet section of the *X86 Quick Guide* for configuration settings for other cards.

CMOS Settings

It may be necessary to modify the BIOS settings in CMOS to boot from a hard disk. Press DEL after rebooting and configure the board with the correct settings for the attached peripherals. The boot sequence should be to try floppy first, and then the IDE disk.

An attached television monitor may be installed by selecting *NTSC* from the TV Output menu in the Chipset Setup window. Refer to the MAUI Graphics Support section of this document for information on configuring OS-9 to use the optional TV Output port.

Installing OS-9 on the STPC System Board

Installation of OS-9 follows these basic steps:

1. Setting the board jumpers and basic CMOS configuration.
2. Making an initial boot floppy using Microware's wizard.
3. Booting OS-9.
4. Loading the hard disk.
5. Modifying the OS-9 boot image to take advantage of the peripherals on the system board.
6. Making the hard disk bootable.
7. Configuring graphics support.

These instructions assume that:

- You have installed Enhanced OS-9 for the STPC.
- You have the board configured with a floppy and hard drive, a monitor and keyboard.
- You have a network connection between the board and the Windows host computer.
- You have a serial connection between the board and the Windows host computer, and have a terminal emulation program, such as Hyperterm, running on your Windows computer. Note, this step is optional, but may be necessary if you would like to move the OS-9 console to a serial port so that graphics applications may have dedicated use of the display/keyboard.

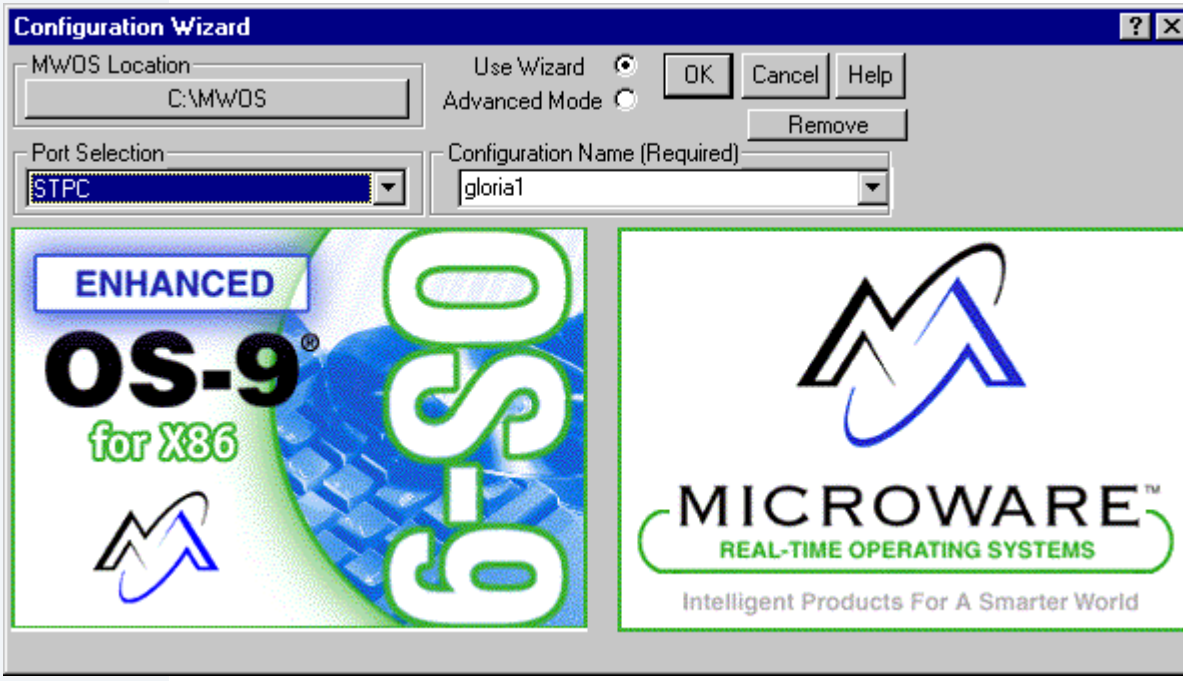
Jumpers and CMOS Configuration

Refer to the previous section for information on the jumper settings.

Making a Boot Floppy

The quickest way to get OS-9 running on your system board is to attach a floppy and use Microware's Configuration Wizard to build a bootable floppy disk.

Once OS-9 is installed on your Windows host computer, select *Microware Configuration Wizard* from the Programs->OS-9 for X86 folder. The following dialog box appears:

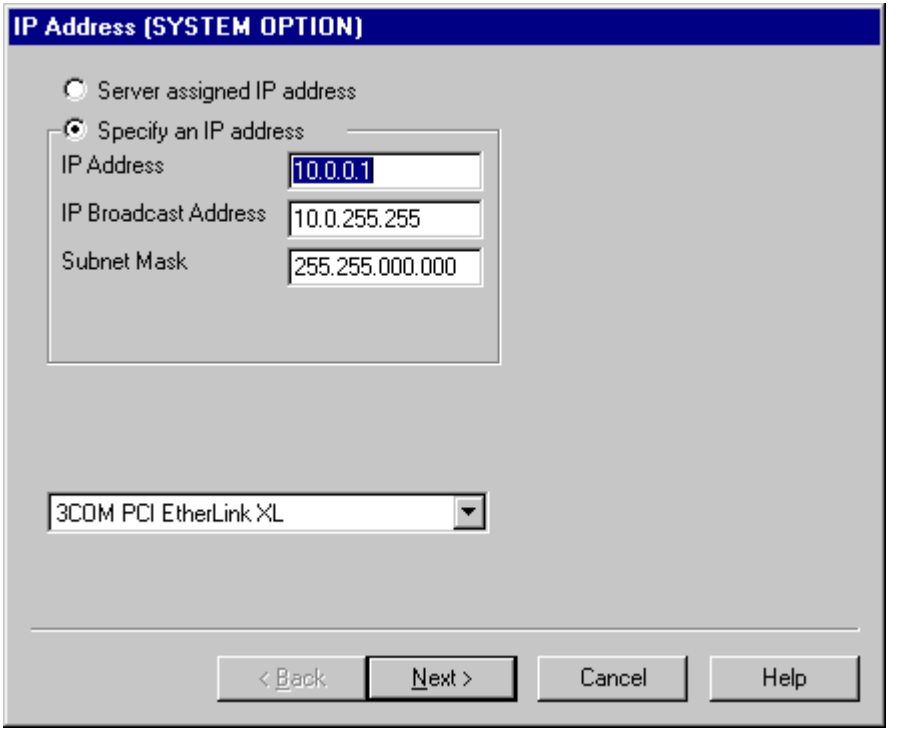


Verify that:

1. The MWOS location reflects the location of the MWOS directory tree installed on your host Windows computer.
2. The *Use Wizard* radio button is selected.
3. The *Port Selection* drop down box displays STPC

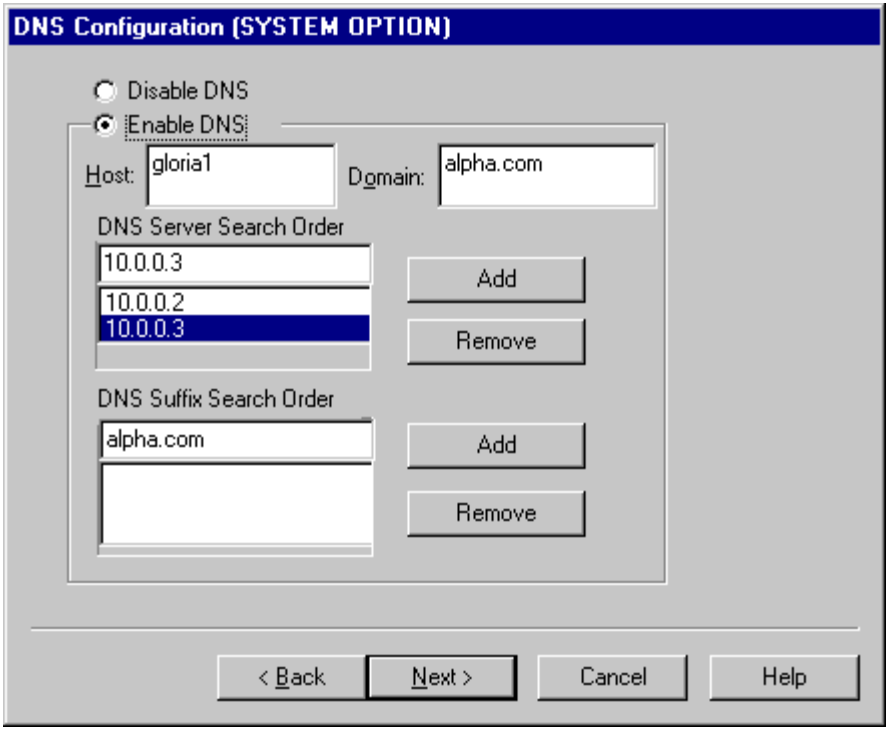
Step 1. Type a name in the *Configuration Name* drop down box (gloria1 in this example) and press *OK*

Step 2. The IP Address Dialog box configures networking for the board. Select the network driver which matches the Ethernet card you have installed, *3COM PCI EtherLink XL* in this example. Select *none* if you will not be using networking.

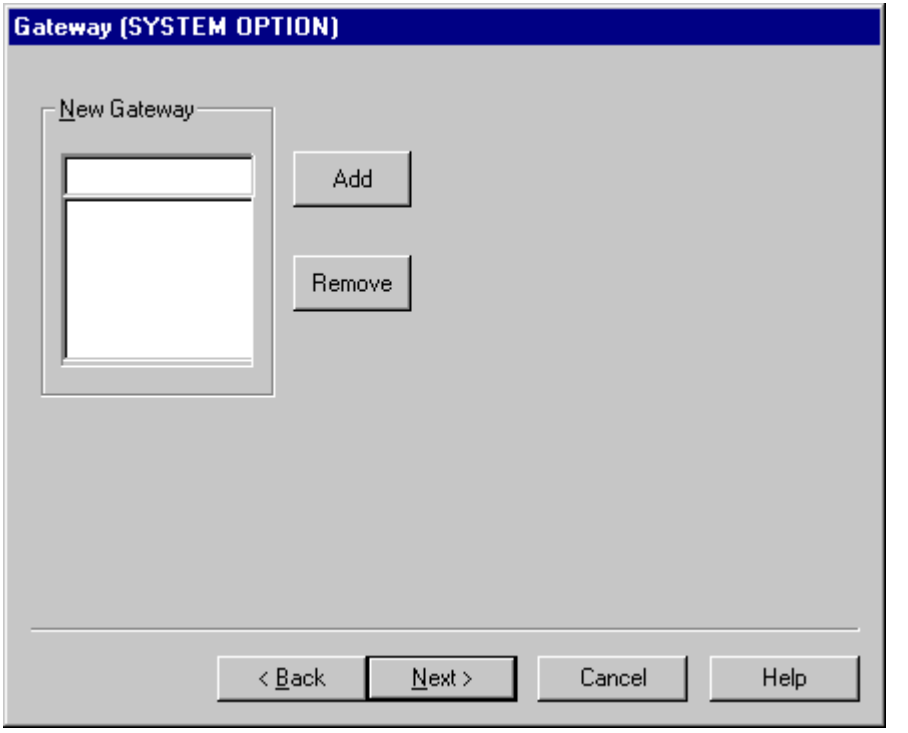


Step 3. Select either the *Server assigned IP address* or the *Specify an IP* address radio button. Enter the IP address, broadcast address and subnet mask, if *Specify an IP address* was selected. Click *Next* to continue.

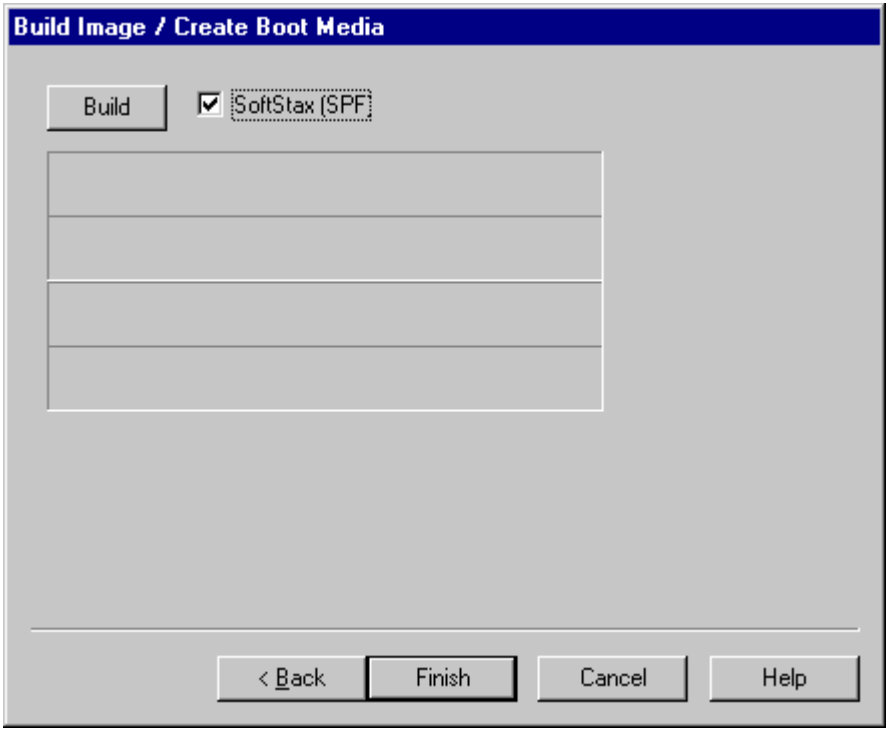
Step 4. Select the *Enable DNS* radio button if DNS is to be used. Fill in the appropriate values for your network. Select *Next* to continue.



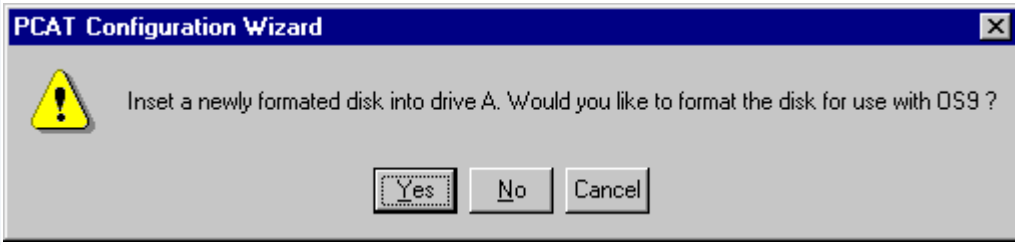
Step 5. Enter the gateway, if appropriate for your network and click *Next* to continue.



Step 6. Enable the *SoftStax (SPF)* check box if using networking. Select the *Build* button to create the OS-9 boot image.



The following dialog box appears.


PCAT Configuration Wizard
Inset a newly formated disk into drive A. Would you like to format the disk for use with OS9 ?
Yes
No
Cancel

Step 7. Insert a new floppy disk into your computer's A drive and click *Yes* to format it for OS-9 and copy the boot image. Select *Finish* and save your changes, when prompted.

The newly created boot floppy may be used to bring up OS-9. The default OS-9 console is a VGA monitor attached to the system board. If networking was enabled, it should be possible to Telnet to the target

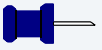
system, from the windows host computer. For example, select *run* from the Windows start menu and type *telnet gloria1*. Login as *super* with the password *user*.

Preparing the Hard Disk

The newly created boot floppy may be used to format a local hard disk with the OS-9 file system. A network connection between the OS-9 target machine and the Windows host computer may be used to load the OS-9 system files onto the hard disk. This section assumes that:

- You have built a boot floppy, with networking enabled, as described in the previous section.
- You have added a hard disk and have used the BIOS setup to configure the disk as the C: drive.
- You have a network connection between the Windows host computer and the OS-9 target system. The target system is configured with an OS-9 supported network interface card.

These instructions step you through copying OS-9 system files to your hard disk. The section, *Finishing the Hard Disk Configuration,* discusses the final steps for making the disk bootable. More information may be found in the *Getting Started with Enhanced OS-9 for X86* manual.



Note

The following instructions assume a minimal configuration with limited disk space (less than 30MB). If additional disk space is available, then you may wish to download mw86.tar in place of mw86sm.tar. mw86.tar includes the full command set and descriptors.

Step 1. Boot the target system, using the floppy made in the previous section.

Step 2. Run fdisk from the OS-9 console.

```
fdisk -d=/hcfmt -e
```

Create OS-9000 type partition.
Make sure the partition is set to active.
Make sure to use MBR option if this is a new disk with no other OS on it.

Step 3. Run format from the OS-9 console to create the OS-9 RBF file system

```
$ format /hc1fmt
```

Note that physical format and physical verifies are typically not necessary.

Step 4. Download the required system files. On the Windows host computer, open a command window and change into the RESIDENT directory on the CD-ROM. Start an ftp session with the target machine.

```
ftp <target>
ftp> User: Super
ftp> Password: User
ftp> bin
ftp> cd /hc1fmt
ftp> send tar
ftp> send diskcache
ftp> send mw86sm.tar
ftp> quit
```

Step 5. Turn on disk cache support, from the OS-9 console.

```
$ chd /hc1fmt ; load -d diskcache ; diskcache -e /hc1fmt=1024k
```

Step 6. Expand the system files.

```
$ load -d tar ; tmode nopause ; tar xvpf mw86sm.tar
```

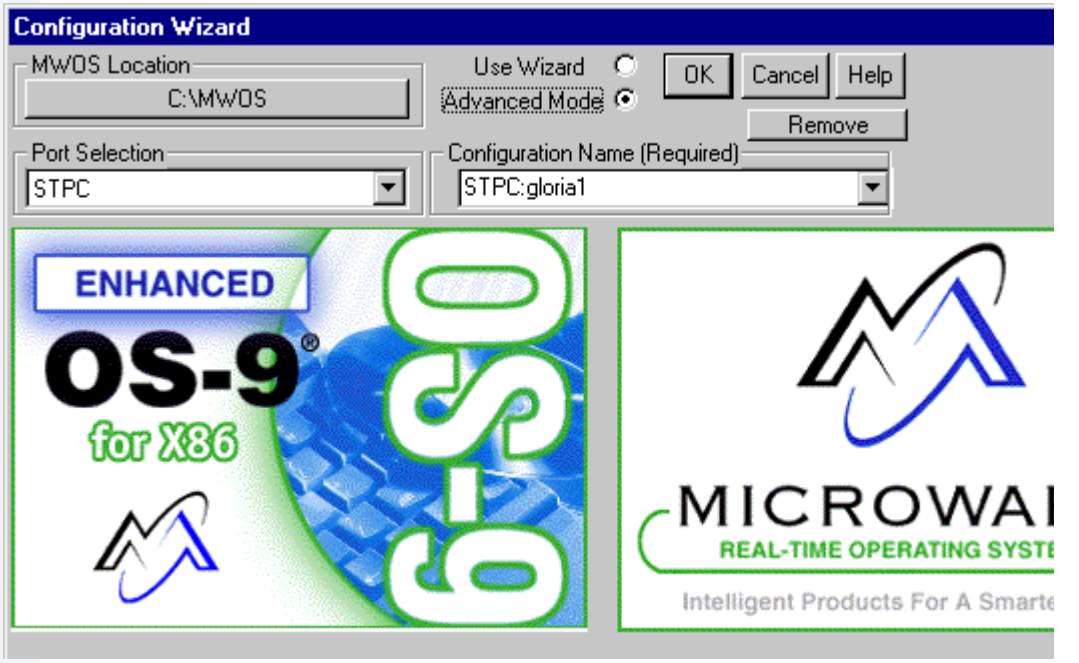
At this point, the disk has been formatted and the OS-9 system files have been copied to disk. The next section, *Advanced Configurations*, discusses building and installing an OS-9 boot file on the hard disk.

Advanced Configurations

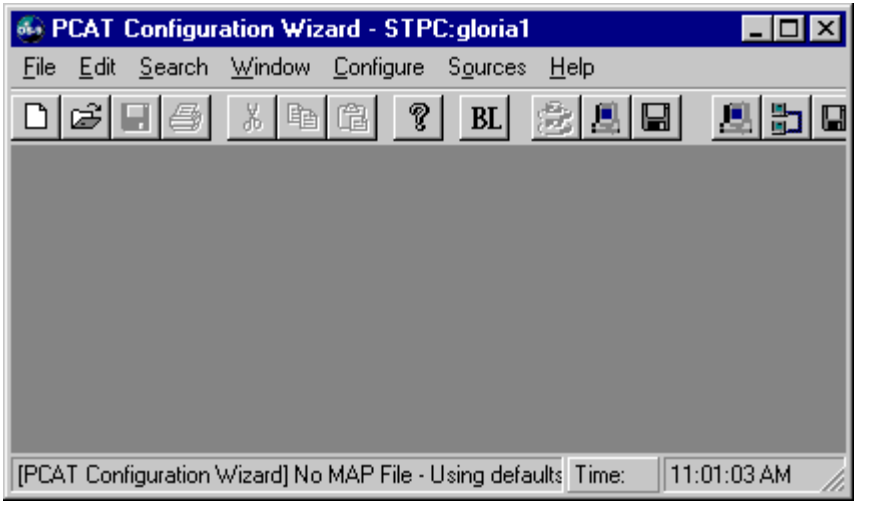
It may be desirable to configure the system to boot from a local hard drive and to move the OS-9 console to a serial port so that the monitor and keyboard may be dedicated to graphics applications. This section assumes that:

- You have built a boot floppy, with networking enabled, as described in the section *Making A Boot Floppy*.
- You have a hard disk attached to the OS-9 target system and have followed the instructions in the previous section to format and load it.
- You have the COM1 port connected to your windows host computer’s serial port, with the appropriate cable, and have a terminal emulation program, such as Hyperterm running. Note, this step is optional, but may be necessary if you would like to move the OS-9 console to a serial port so that graphics applications may have dedicated use of the display/keyboard.

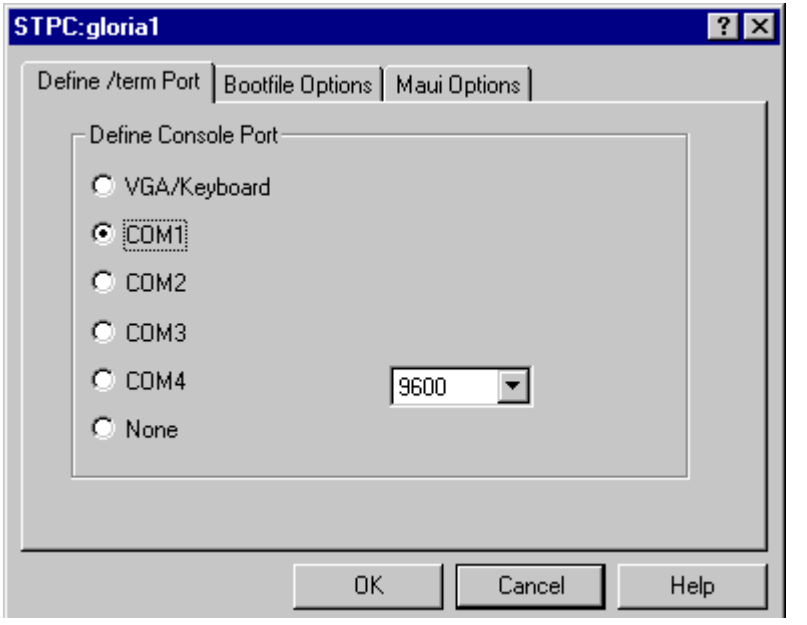
Step 1. Select *Microware configuration Wizard* from the Programs->OS-9 for X86 folder. The following dialog box appears:



Step 2. Click the *Advanced Mode* radio button and select the configuration used when making the boot floppy (gloria1 in this example) from the *Configuration Name* drop down list. Click *OK* to continue.

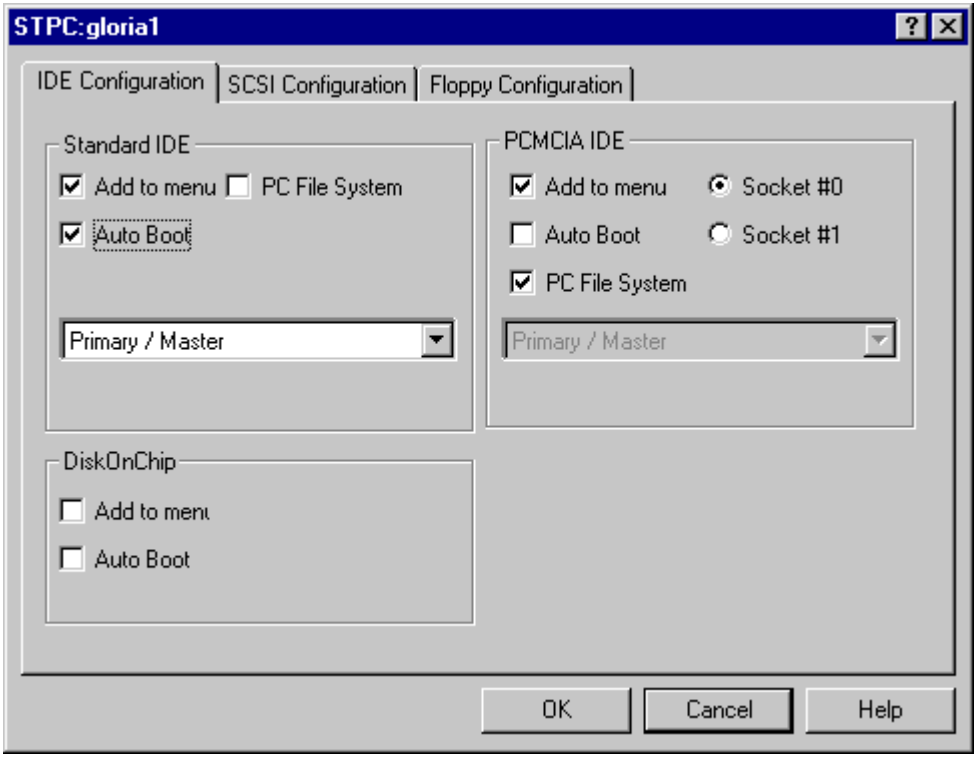

PCAT Configuration Wizard - STPC:gloria1
File Edit Search Window Configure Sources Help
BL
[PCAT Configuration Wizard] No MAP File - Using defaults
Time:
11:01:03 AM

Step 3. Select Configure->Bootfile->Configure System Options from the menu. Click on the *Define /term Port* tab.



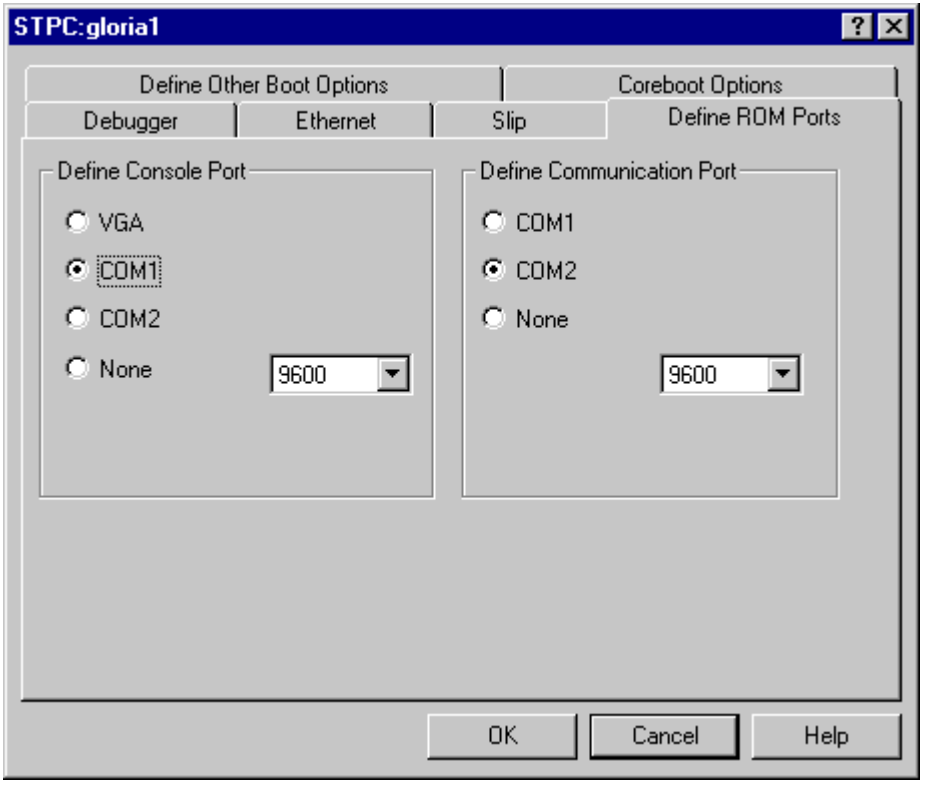
Step 4. Click on the *COM1* radio button to define the high-level console as using serial port 1. Verify that the baud rate is set to 9600 baud. Click *OK* to continue. Note, leave the high-level console at VGA/Keyboard if the monitor and keyboard will **not** be dedicated to graphics applications.

Step 5. Select Configure->Coreboot->Disk Configuration from the menu. Select the *IDE Configuration* tab.



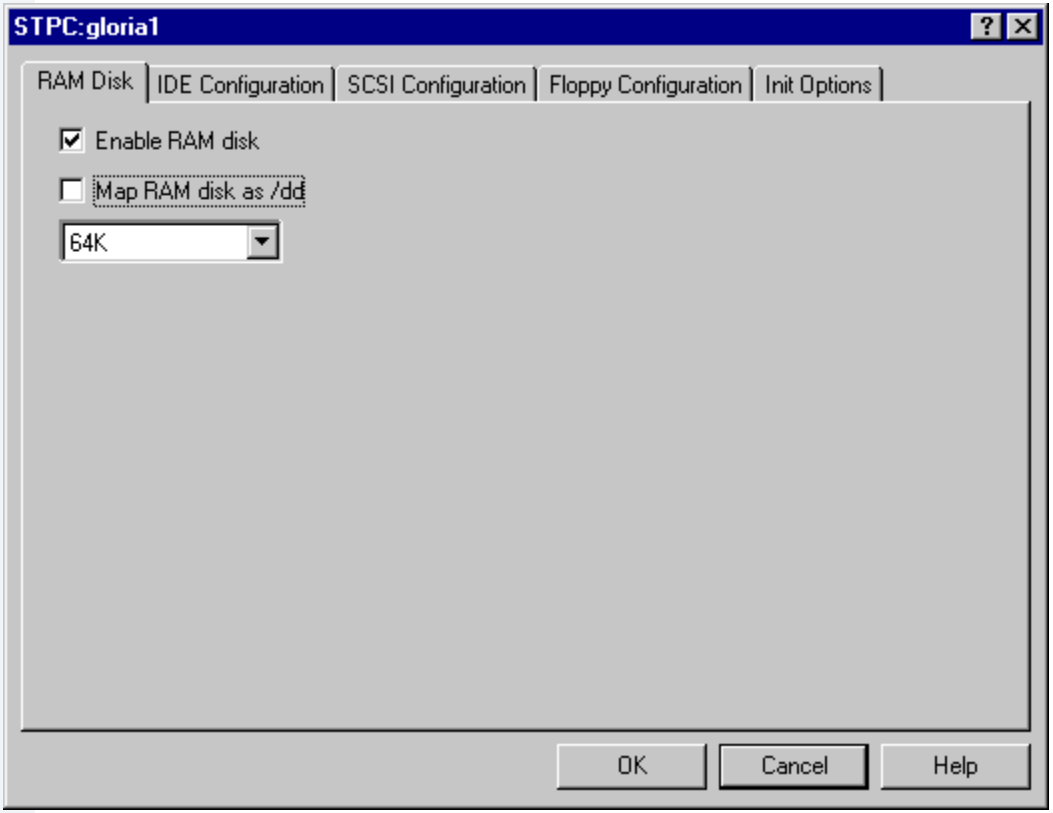
Step 6. Enable the Standard IDE *Auto Boot* checkbox and click *OK* to continue.

Step 7. Select Configure->Coreboot->Main Configuration from the menu. Select the *Define ROM Ports* tab.



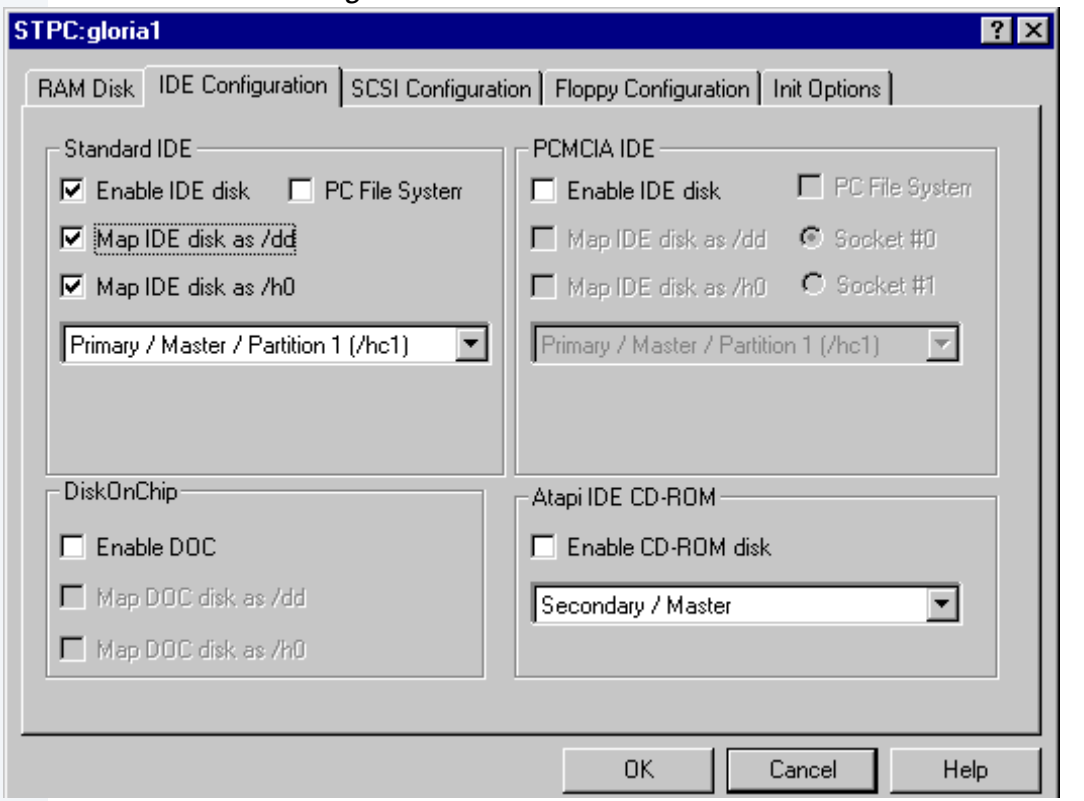
Step 8. Select the Define Console Port *COM1* radio button and verify that the baud rate is set to 9600. Click *OK* to continue. This dialog box moves the low-level OS-9 console to serial port 1. Note, leave the low-level console at VGA if the monitor and keyboard will be used as the OS-9 system console.

Step 9. Select Configure->Bootfile->Disk Configuration from the menu. Click the *RAM Disk* tab.



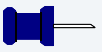
Step 10. Verify that **only** the *Enable RAM disk* checkbox is checked. Select the RAM disk size from the drop down list box. Use of a RAM disk is optional, and you may disable it by clearing the *Enable RAM disk* checkbox.

Step 11. Select the *IDE Configuration* tab.



Step 12. Click on the *Enable IDE disk*, *Map IDE disk as /dd* and *Map IDE disk as /h0* checkboxes to enable them.

Step 13. You may also click on the *Enable IDE disk* checkbox in the PCMCIA IDE window if a PCMCIA controller is attached to the system board. Enable the *PC File System* checkbox when using PCMCIA, ATA devices.



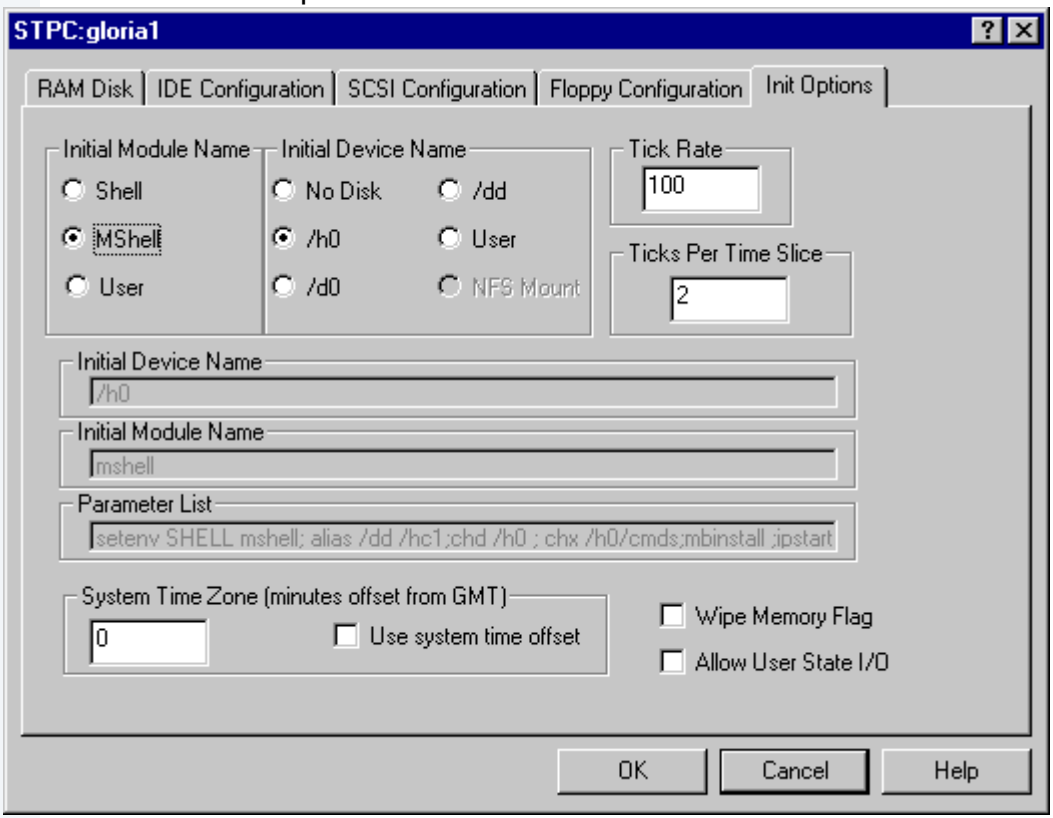
Note

The standard IDE hard disk will be accessed as device /hc1 from the OS-9 console. The same device may also be accessed as /h0 or /dd.

Note

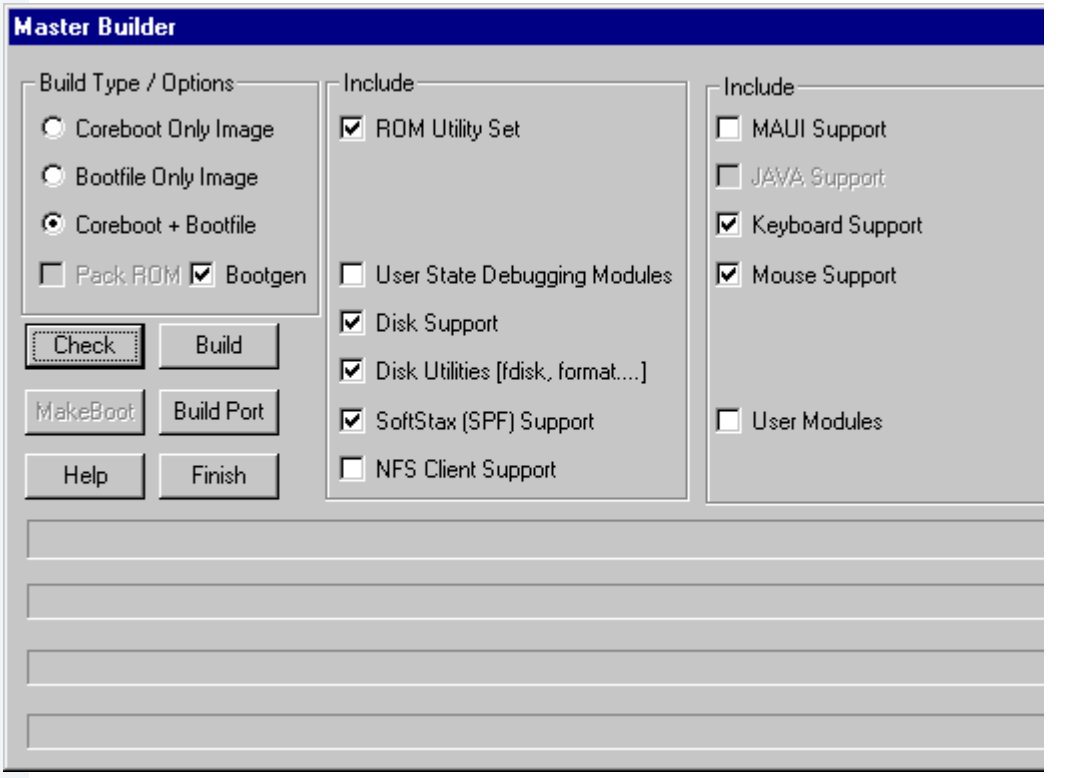
An IDE CD-ROM drive may be attached to the target system and accessed as device /cd0. The CD-ROM must be the master device on the second IDE channel.

Step 14. Click on the Init Options tab.



Step 15. Select the */h0* radio button to use the IDE hard drive as the initial device. Click *OK* to continue.

Step 16. Select Configure->Build Image from the menu. The following window appears.



Step 17. Verify that the following options are enabled:

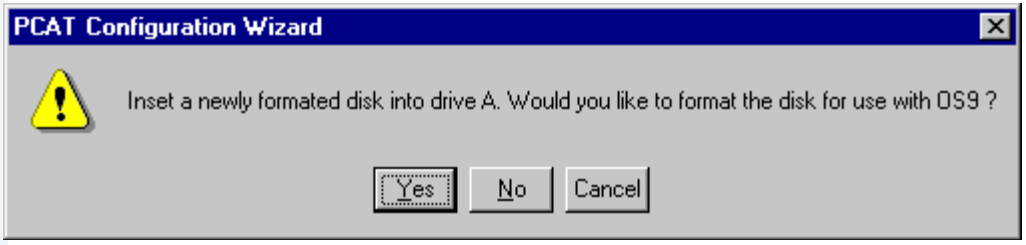
- Coreboot + Bootfile radio button
- Bootgen
- ROM Utility Set
- Enabled Disk Support Modules
- Disk Utilities
- SoftStax (SPF) Support Modules
- Keyboard Support
- Mouse Support

Note

Select the *User State Debugging Modules* checkbox to include the Hawk debugging modules on the target system. Alternately, you may load and run the modules from the hard disk on the target.

Step 18. Click *Build* to create the OS-9 boot image.

Step 19. Click *Makeboot* when the image is built. The following dialog appears.


PCAT Configuration Wizard
Inset a newly formated disk into drive A. Would you like to format the disk for use with OS9 ?
Yes
No
Cancel

Step 20. Insert a new floppy disk into your computer's A drive and click *Yes* to format it for OS-9 and copy the boot image. Select *Finish*, once the copy is complete. Exit the Configuration Wizard by selecting *Exit* from the file menu. Save your changes, when prompted.

Step 21. Test the new boot image by inserting the floppy into the floppy drive on the target system and rebooting. Verify that:

- OS-9 boots, with the console appearing, in the correct location, either on the VGA monitor or on COM1.
- You can telnet to the OS-9 computer, from the Windows computer. For example, select *run* from the Windows start menu and type *telnet gloria1*. Login as *super* with the password *user*.

Finishing the Hard Disk Configuration

This section finishes the hard disk configuration by using the OS-9 bootgen utility to install a boot image onto the hard disk. It is assumed that the instructions in the section *Preparing the Hard Disk* were

followed to format and load the disk. The hard disk will be made bootable using the OS-9 boot image created in the previous section, *Advanced Configurations*.

Step 1. Boot the target system, using the floppy made in the previous section.

Step 2. Verify that you can access the hard disk. For example, from the OS-9 console execute the command: `$ dir /h0`
An OS-9 directory listing should be displayed.

Step 3. Turn disk caching off prior to running bootgen. At the OS-9 console, type the command:

```
$ diskcache -d /hc1fmt
```

Step 4. Bootgen the new system At the OS-9 console, type:

```
$ bootgen /hc1fmt -i=/d0/iplhdnoq -l=/d0/firstboot /d0/sysboot -nb400
```

Step 5. Remove the floppy from the drive and reboot the system. The system should boot from the hard disk, with the OS-9 system prompt appearing on the console.

MAUI Graphics Support

MAUI is Microware's graphics solution. To start MAUI from the OS-9 console, change into the /h0/sys directory and type: loadmaui. Verify that MAUI is running by executing a couple of the demo programs, such as fdraw or fcopy from the OS-9 console.

OS-9 running on the STPC supports 256 colors with 640 x 480, 800 x 600 or 1024 x 768 resolutions. To set the resolution, edit the loadmaui script and find the STPC sections at the end of the file. Uncomment the code block for the desired resolution by removing the leading asterisks. The unused resolution must be commented out.

The STPC Evaluation board may be configured to use the NTSC television output in place of the normal SVGA output by removing the leading asterisks from the NTSC section of the loadmaui file. The unused SVGA resolutions must be commented out. The video output must also be changed in the BIOS CMOS settings.

The following code fragments, from the loadmaui file, configure the display for 640 x 480 resolution.

```
* STPC - Gloria Board
* The gd_???? is for the client version and
* the gd_????_co is for the consumer version
* of the cpu.
* MAUI port - STPC at 640x480 256 color
*
load -d CMDS/BOOTOBJS/MAUI/cdb_stpc
load -d CMDS/BOOTOBJS/MAUI/svga_stpc
load -d CMDS/BOOTOBJS/MAUI/gx_stpc6
load -d CMDS/BOOTOBJS/MAUI/gd_stpc
*load -d CMDS/BOOTOBJS/MAUI/gd_stpc_co
*
* MAUI port - STPC at 800x600 256 color
*
*load -d CMDS/BOOTOBJS/MAUI/cdb_stpc
*load -d CMDS/BOOTOBJS/MAUI/svga_stpc
*load -d CMDS/BOOTOBJS/MAUI/gx_stpc8
*load -d CMDS/BOOTOBJS/MAUI/gd_stpc
*load -d CMDS/BOOTOBJS/MAUI/gd_stpc_co
*
* MAUI port - STPC at 1024x768 256 color
*
*load -d CMDS/BOOTOBJS/MAUI/cdb_stpc
*load -d CMDS/BOOTOBJS/MAUI/svga_stpc
*load -d CMDS/BOOTOBJS/MAUI/gx_stpc1
*load -d CMDS/BOOTOBJS/MAUI/gd_stpc
```

```
*load -d CMDS/BOOTOBJS/MAUI/gd_stpc_co
*
* MAUI port - STPC at 640x480 256 color NTSC output
*
*
*load -d CMDS/BOOTOBJS/MAUI/cdb_stpc
*load -d CMDS/BOOTOBJS/MAUI/svga_stpc
*load -d CMDS/BOOTOBJS/MAUI/gx_stpc6
*load -d CMDS/BOOTOBJS/MAUI/gd_stpcntsc
*load -d CMDS/BOOTOBJS/MAUI/gd_stpcntsc_co
*
* End of bootlist
```

In this example, the file cdb_stpc is the control data block for the mouse. Svga_stpc is the descriptor file for the graphics display. Gx_stpc8 is the 800 x 600 resolution driver, gx_stpc6 is the 640 x 480 resolution driver and gx_stpc1 is the 1024 x 768 resolution driver. Gd_stpc and Gd_stpcntsc are data modules used by the graphic driver.

Cross-hosted Software Development Using Microware Hawk

Microware Hawk is a comprehensive software development and management environment designed to increase the efficiency of development under OS-9. Hawk's highly integrated tool set simplifies and automates the tasks of creating, debugging, analyzing and managing complex real-time software development projects.

Hawk is much more than just an "edit-compile-debug" package. Its expanded functionality addresses all phases of product development – from initial code creation to software version control. This focus on the entire product lifecycle means reduced time-to-market and easier project management.

Microware Hawk is included in the X86 BLS and OEM packages. A fully functional, sixty day evaluation version is included on the OS-9 for X86 evaluation CDs.

This example assumes that a disk based OS-9 system was prepared, as discussed in the previous sections.

Starting Hawk on the Target

Hawk communication and debugging modules must be started on the target, prior to using Hawk. The user state debugging modules may be started from the OS-9 console by changing into the /h0/sys directory on the hard disk and executing startndpd_spf.

```
$ chd /h0/sys
$ startndpd_spf
```

Running MAUI Graphics

The Hawk tutorial requires MAUI graphics. From the OS-9 console, change into the /h0/sys directory and type *loadmaui*, if you have not already done so.

Running the Hawk Tutorial

Included on the CD is a sample MAUI application intended to introduce both MAUI programming and the Hawk IDE to the new OS-9 user. The application, named security, is actually one program in a series of possible applications grouped together to provide a home security suite called HomeWatch.

The purpose of this tutorial is to demonstrate (a) creating a Hawk project file for the MAUI application called HomeWatch, (b) compiling the security application, and (c) running the application on the target.

Project File Creation

In order to create a project file for the HomeWatch system, it is necessary to know where the source code for the application is located. This tutorial assumes all source files are relative to the C:\Mwos directory.

Step 1. Start the Hawk IDE from Windows, by selecting *Microware Hawk IDE* from the Programs->OS-9 for X86 folder

Step 2. When the Hawk IDE appears, go the Project menu and select New->Project. The Create New Project dialog box appears.

Step 3. Type the name of the project in the Name: text field. Let's call this project *HomeWatch*.

Step 4. Change the Default Processor to 80386.

Step 5. Click on the *Next* button.

Step 6. You are now asked for a new component. There will be only one component in this project, the program called security, so type *security* in the Name: text field. You may also want to add a short description of the program, but no other fields need changing. Click on the *Next* button.

Step 7. We must now select the Units that will be associated with this component. Navigate to the C:\MWOS\SRC\MAUI\DEMOS\HOMEWATCH folder, and double click on the security.c source file. This is the only source file that is needed for this program. Click on the *Finish* button.

Once these steps have been completed, Hawk will generate the dependencies for the source file, and you will be presented with the security component on the left side of the Hawk IDE window.

Customizing the Component

Although the project is now created, there is still work needed to customize the security component. Specifically, we need to tell Hawk about the location of 3rd party header files as well as both Microware and 3rd party libraries used to construct the security application.

Step 1. From the Project menu, select *Properties*.

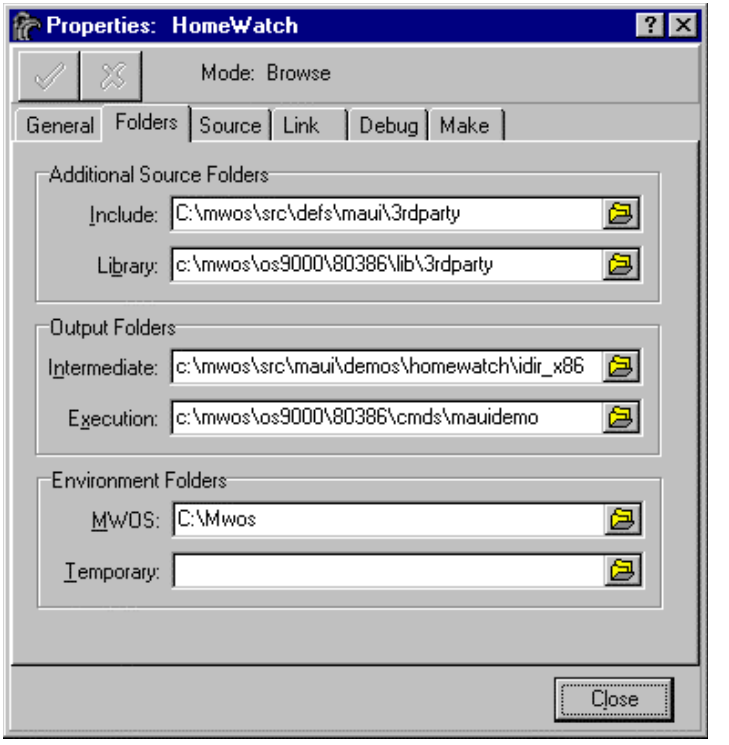
Step 2. When the Properties dialog box appears, select the *Folders* tab. You are presented with 6 text fields.

Step 3. In the Include: text field, type: `C:\MWOS\SRC\DEFS\MAUI\3RDPARTY`

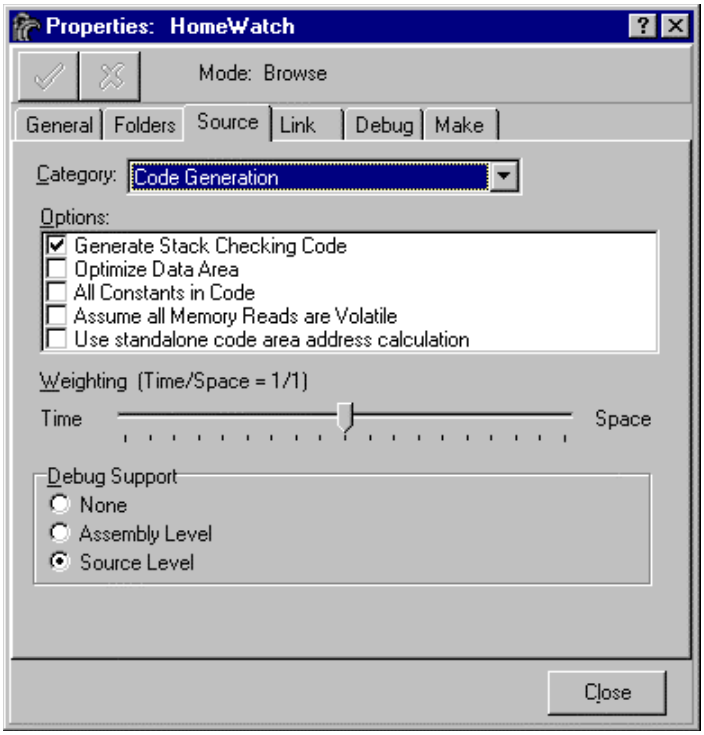
Step 4. In the Library: text field, type: `C:\MWOS\OS9000\80386\LIB\3RDPARTY`

Step 5. In the Intermediate: text field, type: `C:\MWOS\SRC\MAUI\DEMOS\HOMEWATCH\IDIR_X86`

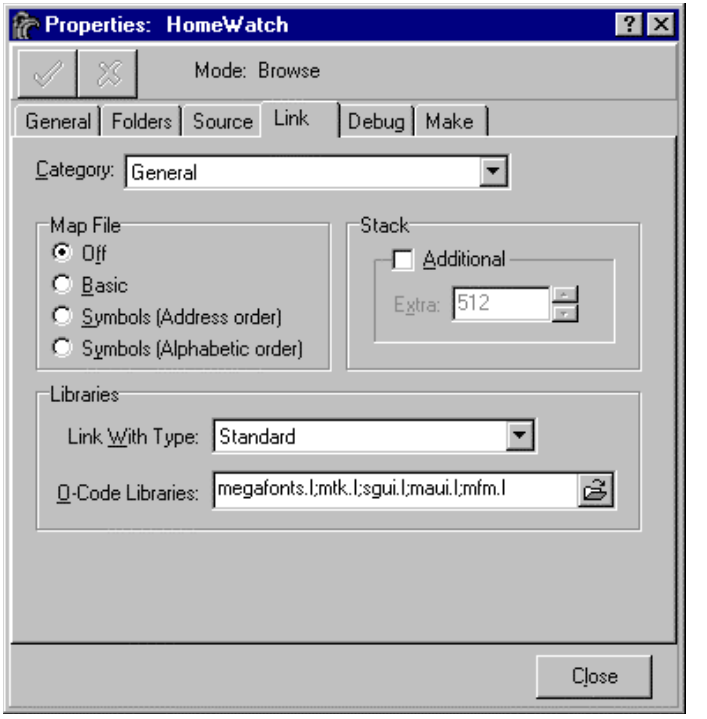
Step 6. In the Execution: text field, type:
`C:\MWOS\OS9000\80386\CMDS\MAUIDEMO`.



Step 7. Select the *Source* Tab. In the Category: list box, select *Code Generation*. Notice the area at the bottom of the dialog box labeled Debug Support. Select the option *Source Level*. This allows us the option of source debugging the application if desired.



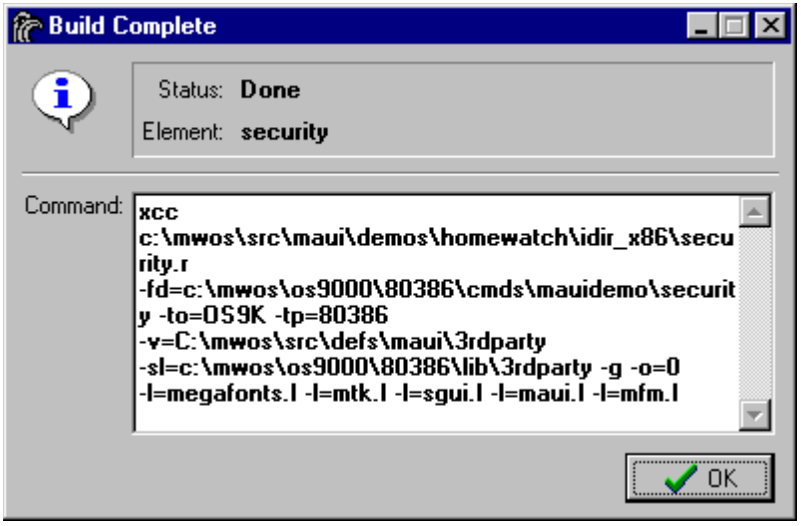
Step 8. Select the *Link* tab. Here we must tell Hawk which libraries to link to. In the O-Code Libraries: text field, type the following:
megafonts.l;mtk.l;sgui.l;maui.l;mfm.l
The library file names must be typed exactly as they appear here, with no embedded spaces.



Step 9. Click on the *Close* button.

Building the Project

Now we are ready to build the project. Go to the Project menu and select *Build*. Hawk will display a dialog box showing the progress of the compilation of the security component. If no errors occur, the dialog box will show Status: Done and you can dismiss the box by clicking the *OK* button. If errors do occur, review the above procedures and be sure you followed them exactly as outlined.



Loading and Executing the HomeWatch Application

To load the application, right click on security in the components box on the left side of the Hawk IDE. When the context box appears, select *Load*. Since this is your first time loading this newly created component, a dialog box will appear asking for the name or IP address of the target to load this component to (gloria1 in the preceding example). Type this information and click on *OK*. The security module will then be loaded to the target

An additional file, which contains graphic assets, must also be loaded.

Step 1. Go the Target menu and select *Load*.

Step 2. In the new dialog box, navigate to the `C:\MWOS\OS9000\80386\CMDS\MAUIDEMO` folder, double click on the file *security_data*, and then click *Load*.

With both the security and security_data files loaded, you can telnet to the target and run security. For example, click on the *Telnet* tab at the bottom of the Hawk window and enter the host name or address. Click on the connect icon and login as *super* with the password *user*. Run the MAUI input process by typing:

```
maui_inp ^255 <>>>/nil&
```

at the OS-9 command prompt and then running the security program by typing *security* at the console prompt.

Security

Upon running the security program, you are presented with a graphic screen containing a keypad. The purpose of this program is to secure a known code and exit, possibly allowing a future HomeWatch application to begin running, or to simply turn off an armed home alarm system.

Try typing in a random code. Upon an unsuccessful entry, you will be notified and allowed to input another code. The correct code is 1-2-4-5-5.

Modifying the Key Code

Changing the length of the key code or even the key code itself is easy. From Hawk, click on security.c on the left hand side of the IDE to edit the source file. Look around line 47 for the following code:

```
#define CODE_LEN 5
static char alarmCode[CODE_LEN] = { '1', '2', '4', '5', '5' };
```

The CODE_LEN definition can be changed to accommodate any code length. Keep in mind that if you change the length, you should also change the proper code in the alarmCode array to match the length.

Once the code has been changed:

Step 1. Save the file (File->Save)

Step 2. Build the security application again by right-clicking on the security component on the left side of the Hawk IDE and selecting *Build*.

Step 3. Once the build has properly completed, we need to be sure the old security application in our target is not running. We must unlink the previously loaded security module by going to the Target menu and selecting *Unlink*. A dialog box appears. Replace the name of the target with your target's IP address or name. Fill in the Module: field with the name security (a link count of 1 should be sufficient to unlink the old security module, so leave this field as is) Click on the Unlink button.

Step 4. Now load the new security module by right-clicking on the security component like before, and selecting *Load*.

Step 5. Run the application again from the telnet session to your target.

Product Discrepancy Report

To: Microware Customer Support

FAX: 515-224-1352

From:_____

Company:_____

Phone:_____

Fax:_____Email:_____

Product Name:

Description of Problem:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Host Platform_____

Target Platform_____